IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application

Applicant(s): Dwyer et al.
Case: 5-13
Serial No.: 09/975,764
Filing Date: October 9, 2001
Group: 2188
Examiner: John A. Lane

Title: Method and Apparatus for Adaptive Cache Frame Locking and Unlocking

---

THRICE CORRECTED APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants hereby submit this corrected Appeal Brief to conform with the current format requirements. The original Appeal Brief was submitted on December 14, 2004 to appeal the final rejection dated August 3, 2004, of claims 1 through 36 of the above-identified patent application.

REAL PARTY IN INTEREST

The present application is assigned to Agere Systems Inc., as evidenced by an assignment recorded on October 9, 2001 in the United States Patent and Trademark Office at Reel 012262, Frame 0653. The assignee, Agere Systems Inc., is the real party in interest.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

STATUS OF CLAIMS

Claims 1 through 36 are pending in the above-identified patent application. Claims 1-36 remain rejected under 35 U.S.C. §103(a) as being unpatentable over the admitted prior art in view of Malamy et al. (United States Patent Number 5,353,425). Claims 1, 5-7, 11, 13, 15, 18-21, 23-25, 27, 29, 31, and 34-36 are being appealed.

STATUS OF AMENDMENTS

There have been no amendments filed subsequent to the final rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

The present invention is directed to a method and apparatus for locking the most recently accessed frames in a cache memory (page 4, lines 1-30).

Independent claim 1 is directed to a cache memory, comprising a plurality of cache frames for storing information from main memory (page 4, lines 1-30); and an adaptive frame locking mechanism (FIG. 1: 100) for locking a number of most recently used frames associated with a task (page 4, lines 1-30; page 6, line 10, to page 7, line 2; page 7, line 20, to page 8, line 13; an integrated circuit comprising the cache memory is also disclosed).

Independent claim 15 is directed to a method for locking frames in a cache memory, the method comprising the steps of storing information from main memory in frames of the cache memory (page 4, lines 1-30); monitoring a number of most recently used frames (page 4, lines 21-30); and locking the number of most recently used frames if a task is interrupted by another task (page 4, lines 1-30; page 6, line 10, to page 7, line 2; page 7, line 20, to page 8, line 13).

Independent claim 23 is directed to a cache memory device comprising, a memory element for storing information from main memory in frames of the cache memory device (page 4, lines 1-30); a monitor for monitoring a number of most recently used frames (page 4, lines 21-30); and an adaptive frame locking mechanism for locking

the number of the most recently used frames if a task is interrupted by another task (page 4, lines 1-30; page 6, line 10, to page 7, line 2; page 7, line 20, to page 8, line 13).

Independent claim 29 is directed to an integrated circuit, comprising: a cache memory having a plurality of cache frames for storing information from main memory (page 4, lines 1-30); and an adaptive frame locking mechanism for locking a number of most recently used frames associated with a task (page 4, lines 1-30; page 6, line 10, to page 7, line 2; page 7, line 20, to page 8, line 13).

Independent claim 33 is directed to a cache memory device comprising: a memory element for storing information from main memory in frames of the cache memory device (page 4, lines 1-30); a monitor for monitoring a number of most recently used frames (page 4, lines 21-30); and an adaptive frame locking mechanism for locking the number of the most recently used frames if a task is interrupted by another task (page 4, lines 1-30; page 6, line 10, to page 7, line 2; page 7, line 20, to page 8, line 13).

In one exemplary embodiment, an identifier of the n most recently used frames is maintained for each of a plurality of tasks (page 4, lines 21-30).

In one exemplary embodiment, all the frames in a set are not locked concurrently (page 5, lines 20-21)

In one exemplary embodiment, a number of the most recently used frames identifies the most recently accessed 3n/2 frames on average (page 6, lines 10-17).

In one exemplary embodiment, an adaptive frame unlocking mechanism automatically unlocks frames that cause a performance degradation for a task (page 7, lines 12-27).

In one exemplary embodiment, the cache is a two way set associative cache and the most recently used frames are identified by taking an inverse of a least recently used identifier (page 4, lines 12-20; and page 8, lines 6-22)

In addition, the present specification discloses means for locking frames that does not lock all the frames in a set concurrently (page 5, lines 20-27), means for unlocking locked frames that automatically unlocks frames that cause a significant performance degradation for a task (page 7, lines 20-27), and means for unlocking that

includes a counter for monitoring a number of times a task experiences a frame miss (page 7, line 20, to page 8, line 5).

STATEMENT OF GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-36 are rejected under 35 U.S.C. §103(a) as being unpatentable over the admitted prior art in view of Malamy et al.

ARGUMENT

Independent claims 1, 15, 23, and 29 are rejected under 35 U.S.C. §103(a) as being unpatentable over the admitted prior art in view of Malamy et al.

In particular, the Examiner asserts that the admitted prior art teaches the claimed step of "locking frames if a task is interrupted by another task." The Examiner acknowledges that the admitted prior art does not discuss locking a frame or frames in accordance with a most recently used scheme, but asserts that Malamy teaches locking pages or blocks in the cache in accordance with a most recently used locking scheme.

First, Applicants note that the admitted prior art teaches to lock all frames associated with a task, if the task is interrupted by another task. Independent claims 1 and 29 require locking a number of most recently used frames *associated with a task*. Independent claims 15 and 23 require locking said number of said most recently used frames *if a task is interrupted by another task*. Thus, the admitted prior art actually *teaches away* from the present invention by teaching to lock ***all*** frames associated with a task.

Applicants also note that Malamy teaches a scheme that prevents the most recently used lines in a cache from being replaced when the cache controller is forced to replace a cache memory line. The most recently used cache lines are thus blocked from being replaced, *regardless of the task they are associated with and regardless of whether a task is interrupted by another task*. The present invention, alternatively, recognizes that the most recently accessed frames in a cache memory are likely to be accessed by a task again in the near future. Thus, the most recently used frames *associated with a task* may be locked in accordance with the present invention at the beginning of a task switch or

interrupt, and are thus available when an interrupted task resumes execution (to improve the performance of the cache. Independent claims 1 and 29 require locking a number of most recently used frames *associated with a task*. Independent claims 15 and 23 require locking said number of said most recently used frames *if a task is interrupted by another task*. Malamy, therefore, actually *teaches away* from the present invention by teaching to block the replacement of the most recently used cache lines ***regardless of the task they are associated with***.

Thus, the admitted prior art and Malamy, alone or in combination, do not disclose or suggest locking a number of most recently used frames associated with a task, as required by independent claims 1 and 29, and do not disclose or suggest locking said number of said most recently used frames if a task is interrupted by another task, as required by independent claims 15 and 23. Furthermore, Applicants could find no disclosure or suggestion in the prior art to combine the prior art techniques cited by the Examiner and, as stated above, each of the cited prior art disclosures actually teaches away from the present invention. Thus, a person of ordinary skill in the art would not look to combine Malamy and the admitted prior art.

Claims 5-7, 11, 13, 18-21, 24, 25, 27, 31 and 34-36

Claims 5/18, 6/19/24/34, 7/20, 11/21/25/31/35, and 13/27/36 specify a number of limitations providing additional bases for patentability. Specifically, the Examiner rejected the cited claims under 35 U.S.C. §103(a) as being unpatentable over the admitted prior art in view of Malamy et al. Claims 5 and 18 require an identifier of the n most recently used frames is maintained for each of a plurality of tasks. Claims 6, 19, 24, and 34 require not locking all the frames in a set concurrently. Claims 7 and 20 require wherein said number of said most recently used frames identifies the most recently accessed 3n/2 frames on average. Claims 11, 25, 31, and 35 require an adaptive frame unlocking mechanism that automatically unlocks frames that cause a performance degradation for a task. Claims 13 and 36 require wherein said cache is a two way set associative cache and said most recently used frames are identified by taking an inverse of a least recently used identifier.

Regarding the dependent claims, the Examiner asserts that it is believed that most, if-not-all, dependent claim features are taught by the admitted prior art and/or Malamy.

The admitted prior art and Malamy (alone or in combination), however, do not disclose or suggest an identifier of the n most recently used frames is maintained for each of a plurality of tasks, as required by claims 5 and 18, do not disclose or suggest not locking all the frames in a set concurrently, as required by claims 6, 19, 24, and 34, do not disclose or suggest wherein said number of said most recently used frames identifies the most recently accessed 3n/2 frames on average, as required by claims 7 and 20, do not disclose or suggest an adaptive frame unlocking mechanism that automatically unlocks frames that cause a performance degradation for a task, as required by claims 11, 25, 31, and 35, and do not disclose or suggest wherein said cache is a two way set associative cache and said most recently used frames are identified by taking an inverse of a least recently used identifier, as required by claims 13 and 36.

Conclusion

The rejections of the cited claims under section §103 in view of the admitted prior art and Malamy are therefore believed to be improper and should be withdrawn. The remaining rejected dependent claims are believed allowable for at least the reasons identified above with respect to the independent claims.

The attention of the Examiner and the Appeal Board to this matter is appreciated.

Respectfully,

Kevin M. Mason

Date: August 31, 2007

Kevin M. Mason
Attorney for Applicant(s)
Reg. No. 36,597
Ryan, Mason & Lewis, LLP
1300 Post Road, Suite 205
Fairfield, CT 06824
(203) 255-6560

APPENDIX

1. A cache memory, comprising:

a plurality of cache frames for storing information from main memory; and

an adaptive frame locking mechanism for locking a number of most recently used frames associated with a task.

2. The cache memory of claim 1, further comprising a memory for recording an identifier of the n most recently used frames

3. The cache memory of claim 2, wherein said identifier is a frame address.

4. The cache memory of claim 2, wherein said identifier is a flag associated with said most recently used frames.

5. The cache memory of claim 2, wherein said identifier of the n most recently used frames is maintained for each of a plurality of tasks.

6. The cache memory of claim 1, wherein said adaptive frame locking mechanism does not lock all the frames in a set concurrently.

7. The cache memory of claim 1, wherein said number of said most recently used frames identifies the most recently accessed 3n/2 frames on average

8 The cache memory of claim 1, wherein said adaptive frame locking mechanism includes three latches (a, b, and lock) for each frame of said cache.

9. The cache memory of claim 8, wherein said latch a is set when a frame is accessed and the value in latch a of a frame is transferred to latch b and latch a is reset after n accesses.

10. The cache memory of claim 8, wherein said adaptive frame locking mechanism sets a lock latch of a given frame, locking the frame, if either latch a or latch b is set when the lock signal is asserted.

11. The cache memory of claim 1, further comprising an adaptive frame unlocking mechanism that automatically unlocks frames that cause a performance degradation for a task.

12. The cache memory of claim 11, wherein said adaptive frame unlocking mechanism includes a counter for monitoring a number of times a task experiences a frame miss.

13. The cache memory of claim 1, wherein said cache is a two way set associative cache and said most recently used frames are identified by taking an inverse of a least recently used identifier.

14. The cache memory of claim 1, wherein said locking is performed if a first task is interrupted by a second task.

15. A method for locking frames in a cache memory, said method comprising the steps of:

storing information from main memory in frames of said cache memory;

monitoring a number of most recently used frames; and

locking said number of said most recently used frames if a task is interrupted by another task.

16. The method of claim 15, wherein said monitoring step maintains a frame address of said most recently used frames.

17. The method of claim 15, wherein said monitoring step maintains a flag associated with said most recently used frames.

18. The method of claim 15, wherein said monitoring step maintains an identifier of the n most recently used frames for each of a plurality of tasks.

19. The method of claim 15, wherein said locking step does not lock all the frames in a set concurrently.

20. The method of claim 15, wherein said number of said most recently used frames identifies the most recently accessed 3n/2 frames on average.

21. The method of claim 15, further comprising the step of automatically unlocking frames that cause a significant performance degradation for a task.

22. The method of claim 21, wherein said step of unlocking further comprises the step of monitoring a number of times a task experiences a frame miss.

23. A cache memory comprising:

a memory element for storing information from main memory in frames of said cache memory;

means for monitoring a number of most recently used frames; and

means for locking said number of said most recently used frames if a task is interrupted by another task.

24. The cache memory of claim 23, wherein said means for locking said frames does not lock all the frames in a set concurrently.

25. The cache memory of claim 23, further comprising means for unlocking said locked frames that automatically unlocks frames that cause a significant performance degradation for a task.

26. The cache memory of claim 25, wherein said means for unlocking includes a counter for monitoring a number of times a task experiences a frame miss.

27. The cache memory of claim 23, wherein said cache is a two way set associative cache and said most recently used frames are identified by taking an inverse of a least recently used identifier.

28. The cache memory of claim 23, wherein said locking is performed if a first task is interrupted by a second task.

29. An integrated circuit, comprising:

a cache memory having a plurality of cache frames for storing information from main memory; and

an adaptive frame locking mechanism for locking a number of most recently used frames associated with a task.

30. The integrated circuit of claim 29, further comprising a memory for recording an identifier of the n most recently used frames.

31. The integrated circuit of claim 29, further comprising an adaptive frame unlocking mechanism that automatically unlocks frames that cause a performance degradation for a task

32. The integrated circuit of claim 29, wherein said locking is performed if a first task is interrupted by a second task.

33. A cache memory device comprising:

a memory element for storing information from main memory in frames of said cache memory device;

a monitor for monitoring a number of most recently used frames; and

an adaptive frame locking mechanism for locking said number of said most recently used frames if a task is interrupted by another task.

34. The cache memory device of claim 33, wherein said adaptive frame locking mechanism does not lock all the frames in a set concurrently

35. The cache memory device of claim 33, wherein said adaptive frame locking mechanism automatically unlocks frames that cause a significant performance degradation for a task.

36. The cache memory device of claim 33, wherein said cache is a two way set associative cache and said most recently used frames are identified by taking an inverse of a least recently used identifier.

EVIDENCE APPENDIX

There is no evidence submitted pursuant to § 1.130, 1 131, or 1.132 or entered by the Examiner and relied upon by appellant.

RELATED PROCEEDINGS APPENDIX

There are no known decisions rendered by a court or the Board in any proceeding identified pursuant to paragraph (c)(1)(ii) of 37 CFR 41.37.